CLAIMS

1. In a cluster of computing nodes having shared access to one or more volumes of data storage using a parallel file system, a method for managing the data storage, comprising:

initiating a session of a data management application on a session node selected from among the nodes in the cluster;

receiving an event message in a session queue for processing by the data management application at the session node, responsive to a request submitted to the parallel file system by a user application on a source node among the nodes in the cluster to perform a file operation on a file in the data storage; and

following a failure at the session node, reconstructing the session queue so that processing of the event message by the data management application can continue after recovery from the failure.

2. A method according to claim 1, wherein the failure at the session node comprises a file system failure at the session node.

3. A method according to claim 2, wherein reconstructing the session queue comprises selecting a new session node from among the nodes on which the file system failure has not occurred, and moving the data management session to the new session node.

4. A method according to claim 1, wherein reconstructing the session queue comprises selecting a new session node from among the nodes in the cluster, and assuming the data management session on the new session

node, whereupon the session queue is reconstructed on the new session node.

5. A method according to claim 4, wherein assuming the data management session comprises moving the session to a different node from the session node used before the failure.

6. A method according to claim 4, wherein assuming the data management session comprises assuming the session on the same session node that was used before the failure.

7. A method according to claim 6, wherein assuming the session comprises explicitly invoking a session creation function call of a data management application programming interface (DMAPI).

8. A method according to claim 6, wherein the failure comprises a file system failure at the session node, which is followed by file system recovery, and wherein assuming the session comprises invoking any function call of a data management application programming interface (DMAPI) at the session node after the recovery, whereby reconstruction of the session queue is triggered implicitly.

9. A method according to claim 1, and comprising storing information regarding the session and events before the failure at one or more additional nodes among the nodes in the cluster, wherein reconstructing the session queue comprises using the information stored at the one or more additional nodes to reconstruct the queue.

10. A method according to claim 1, and comprising selecting one of the nodes to serve as a session manager

node, and assuming the session by sending a message to the session manager node, causing the session manager node to distribute information regarding the session among the nodes in the cluster so that the data management application can continue after the recovery.

11. A method according to claim 1, wherein initiating the session comprises initiating the session in accordance with a data management application programming interface (DMAPI) of the parallel file system, and wherein processing the event message comprises processing the request using the DMAPI.

12. A method according to claim 1, and comprising:

sending a response to the event message from the data management application on the session node to the source node following the recovery from the failure; and

performing the file operation requested by the source node subject to the response from the data management application.

13. A method according to claim 12, wherein receiving the event message comprises receiving the message responsive to submission of the request by a file operation thread of a user application running on the source node, and blocking the thread until the response is received from the session node after the recovery from the failure.

14. A method according to claim 13, wherein reconstructing the session queue comprises sending a message from the session node to all of the nodes, so as to prompt the file operation thread on the source node to submit a new event message to the session node, whereby

the event is placed in the reconstructed queue responsive to the new message.

15. A method according to claim 14, wherein prompting the file operation thread comprises instructing the file operation thread to submit the new event message with respect to an event that is defined as a synchronous event.

16. A method according to claim 15, wherein an event that is defined as an asynchronous event that was in the session queue prior to the failure is not placed in the reconstructed queue.

17. A method according to claim 14, wherein receiving the event message comprises receiving an event identifier, which is assigned to the event at the source node, and wherein the event placed in the reconstructed queue has the same event identifier as was assigned before the failure.

18. A method according to claim 1, and comprising processing the event message in the reconstructed queue, and responsive to the event message, reacquiring a data management access right needed to handle the request.

19. A method according to claim 1, wherein receiving the event message comprises receiving multiple event messages from multiple source nodes in the cluster, and wherein reconstructing the session queue comprises collecting information regarding the session and events from the multiple source nodes.

20. A method according to claim 1, wherein initiating the session of the data management application comprises initiating a data migration application, so as to free

storage space on at least one of the volumes of data storage.

21. A method according to claim 1, and comprising, following the failure, when the source node has not received a response to the event message within a predetermined lapse of time, failing the request submitted at the source node to the parallel file system.

22. Computing apparatus, comprising:

one or more volumes of data storage, arranged to store data; and

a plurality of computing nodes, linked to access the volumes of data storage using a parallel file system, and arranged so as to enable a data management application to initiate a data management session on a session node selected among the nodes in the cluster, so that when a request is submitted to the parallel file system by a user application on a source node among the nodes in the cluster to perform a file operation on a file in the data storage, an event message is received at the session node responsive to the request, for processing by the data management application, and so that following a failure at the session node, the session queue is reconstructed so that processing of the event message by the data management application can continue after recovery from the failure.

23. Apparatus according to claim 22, wherein the failure at the session node comprises a file system failure at the session node.

24. Apparatus according to claim 23, wherein the nodes are arranged so that following the file system failure, a

new session node is selected from among the nodes on which the file system failure has not occurred, and the data management session is moved to the new session node.

25. Apparatus according to claim 22, wherein the nodes are arranged so that following the failure, a new session node is selected from among the nodes on which the failure has not occurred, and the data management session is assumed on the new session node, whereupon the session queue is reconstructed on the new session node.

26. Apparatus according to claim 25, wherein the session is assumed on a different node from the session node used before the failure.

27. Apparatus according to claim 25, wherein the session is assumed on the same session node that was used before the failure.

28. Apparatus according to claim 27, wherein the session is assumed by explicitly invoking a session creation function call of a data management application programming interface (DMAPI).

29. Apparatus according to claim 27, wherein the failure comprises a file system failure at the session node, which is followed by file system recovery, and wherein the session is assumed by invoking any function call of a data management application programming interface (DMAPI) at the session node after the recovery, whereby reconstruction of the session queue is triggered implicitly.

30. Apparatus according to claim 22, wherein the nodes are arranged so that information regarding the session and events is stored before the failure at one or more

additional nodes among the nodes in the cluster, whereby the session queue is reconstructed using the information stored at the one or more additional nodes.

31. Apparatus according to claim 22, wherein one of the nodes is selected to serve as a session manager node, and wherein to assume the session, a message is sent to the session manager node, causing the session manager node to distribute information regarding the session among the nodes in the cluster so that the data management application can continue after the recovery.

32. Apparatus according to claim 22, wherein the session is initiated in accordance with a data management application programming interface (DMAPI) of the parallel file system, and wherein the event message is processed using the DMAPI.

33. Apparatus according to claim 22, wherein the nodes are arranged so that a response to the event message is sent from the data management application on the session node to the source node following the recovery from the failure, whereupon the file operation requested by the source node is carried out subject to the response from the data management application.

34. Apparatus according to claim 33, wherein the event message is received responsive to submission of the request by a file operation thread of a user application running on the source node, and the thread is blocked until the response is received from the session node after the recovery from the failure.

35. Apparatus according to claim 34, wherein to reconstruct the session queue, a message is sent from the

session node to all of the nodes, so that the file operation thread on the source node is prompted to submit a new event message to the session node, whereby the event is placed in the reconstructed queue responsive to the new message.

36. Apparatus according to claim 35, wherein the file operation thread is prompted to submit the new event message with respect to an event that is defined as a synchronous event.

37. Apparatus according to claim 36, wherein an event that is defined as an asynchronous event that was in the session queue prior to the failure is not placed in the reconstructed queue.

38. Apparatus according to claim 35, wherein the event message contains an event identifier, which is assigned to the event at the source node, and wherein the event placed in the reconstructed queue has the same event identifier as was assigned before the failure.

39. Apparatus according to claim 22, wherein after reconstructing the session queue, the data management application reacquires a data management access right needed to handle the request.

40. Apparatus according to claim 22, wherein the nodes are arranged so that the session node receives multiple event messages from multiple source nodes in the cluster, and so that in reconstructing the session queue, information regarding the session and events is collected from the multiple source nodes.

41. Apparatus according to claim 22, wherein the data management application comprises a data migration

application, for freeing storage space on at least one of the volumes of data storage.

42. Apparatus according to claim 22, wherein following the failure, when the source node has not received a response to the event message within a predetermined lapse of time, the request submitted at the source node to the parallel file system is failed.

43. A computer software product for use in a cluster of computing nodes having shared access to one or more volumes of data storage using a parallel file system, the product comprising a computer-readable medium in which program instructions are stored, which instructions, when read by the computing nodes, cause a session of a data management application to be initiated on a session node selected among the nodes in the cluster, such that when a user application on a source node among the nodes in the cluster submits a request to the parallel file system to perform a file operation on a file in the data storage, an event message is received at the session node, for processing by the data management application, and such that following a failure at the session node, the session queue is reconstructed so that processing of the event message by the data management application can continue after recovery from the failure.

44. A product according to claim 43, wherein the failure at the session node comprises a file system failure at the session node.

45. A product according to claim 44, wherein following the file system failure, the instructions cause a new session node to be selected from among the nodes on which

the file system failure has not occurred, whereupon the data management session is moved to the new session node.

46. A product according to claim 43, wherein following the failure, the instructions cause a new session node to be selected from among the nodes on which the failure has not occurred, whereupon the data management session is assumed on the new session node, and the session queue is reconstructed on the new session node.

47. A product according to claim 46, wherein the session is assumed on a different node from the session node used before the failure.

48. A product according to claim 46, wherein the session is assumed on the same session node that was used before the failure.

49. A product according to claim 48, wherein the session is assumed by explicitly invoking a session creation function call of a data management application programming interface (DMAPI).

50. A product according to claim 48, wherein the failure comprises a file system failure at the session node, which is followed by file system recovery, and wherein the session is assumed by invoking any function call of a data management application programming interface (DMAPI) at the session node after the recovery, whereby reconstruction of the session queue is triggered implicitly.

51. A product according to claim 43, wherein the instructions cause information regarding the session and events to be stored before the failure at one or more additional nodes among the nodes in the cluster, whereby

the session queue is reconstructed using the information stored at the one or more additional nodes.

52. A product according to claim 43, wherein one of the nodes is selected to serve as a session manager node, and wherein to assume the session, a message is sent to the session manager node, causing the session manager node to distribute information regarding the session among the nodes in the cluster so that the data management application can continue after the recovery.

53. A product according to claim 43, wherein the product comprises a data management application programming interface (DMAPI) of the parallel file system, and wherein the event message is processed using the DMAPI.

54. A product according to claim 43, wherein the instructions cause a response to the event message to be sent from the data management application on the session node to the source node following the recovery from the failure, whereupon the file operation requested by the source node is carried out subject to the response from the data management application.

55. A product according to claim 54, wherein the event message is received responsive to submission of the request by a file operation thread of a user application running on the source node, and the thread is blocked until the response is received from the session node after the recovery from the failure.

56. A product according to claim 55, wherein to reconstruct the session queue, a message is sent from the session node to all of the nodes, so that the file operation thread on the source node is prompted to submit

a new event message to the session node, whereby the event is placed in the reconstructed queue responsive to the new message.

57. A product according to claim 56, wherein the file operation thread is prompted to submit the new event message with respect to an event that is defined as a synchronous event.

58. A product according to claim 57, wherein an event that is defined as an asynchronous event that was in the session queue prior to the failure is not placed in the reconstructed queue.

59. A product according to claim 56, wherein the event message contains an event identifier, which is assigned to the event at the source node, and wherein the event placed in the reconstructed queue has the same event identifier as was assigned before the failure.

60. A product according to claim 43, wherein after reconstructing the session queue, the data management application reacquires a data management access right needed to handle the request.

61. A product according to claim 43, wherein the instructions cause the session node to receive multiple event messages from multiple source nodes in the cluster, and to reconstruct the session queue by collecting information regarding the session and events from the multiple source nodes.

62. A product according to claim 43, wherein the data management application comprises a data migration application, for freeing storage space on at least one of the volumes of data storage.

63. A product according to claim 43, wherein following the failure, when the source node has not received a response to the event message within a predetermined lapse of time, the request submitted at the source node to the parallel file system is failed.